

oprintanswers

Production de documents avec L^AT_EX

Variables et Commandes Les variables et commandes L^AT_EX commencent toujours par le caractère \. Il existe de nombreuses variables et commandes prédéfinies. On peut définir (redéfinir) des variables et des commandes. De nombreuses variables définissent les caractéristiques des pages à produire comme `\textheight`, `\textwidth`, `\topmargin`, ...

Commentaires Est considéré comme commentaire, tout texte situé à droite du caractère %.

Quotation La plupart des caractères spéciaux peuvent être quotés en les précédant par le caractère \. Quelques exceptions : le caractère ~ est obtenu grâce à `\textasciitilde`, le caractère \ grâce à `\textbackslash`.

1 Premier document

Sous Emacs tapez votre premier document L^AT_EX.

```
\documentclass{article}

\begin{document}
Ceci est mon \textbf{premier} document en \LaTeX,
écrit le \today en TP d'environnement de programmation,
avec le mode « article ».
```

Après une ligne blanche, commence un nouveau paragraphe dont la première ligne est `\emph{indentée}`.

```
\end{document}
```

Compilez ce fichier avec la commande

```
latex premier
```

Le résultat est dans le fichier `premier.dvi` visualisable grâce à `xdvi` :

```
xdvi premier &
```

On peut obtenir un document au format PostScript en utilisant la commande `dvips` :

```
dvips premier -o premier.ps
```

```
dvips premier -o
```

puis visionnez le document PostScript au moyen de la commande `gv` :

```
gv premier.ps &
```

Remarquez qu'il y a un problème avec les caractères accentués, les guillemets français (« et »), et que la date est en anglais. Pour pouvoir utiliser directement des caractères `isolatin-1` et être dans un environnement adapté au français (espacement correct, date en français, ...), il faut utiliser les paquets suivants :

```
\usepackage[latin1]{inputenc}
```

```
\usepackage[T1]{fontenc}
```

```
\usepackage[francais]{babel}
```

Rajoutez ces trois lignes après la première ligne. Remarquez qu'il n'y a pas d'espace après la date. Dans le texte source, le caractère espace qui suit le nom de la commande `\today` délimite la fin de la commande. Remplacez `\today` par `\today{ }`; recompilez et vérifiez le résultat obtenu.

Remarque. Si on ne peut pas être en permanence en mode « accent », on peut mettre localement les accents `\'e` pour é, `\'a` pour à, `\^i` pour î, `\cc` pour ç, etc.

1.1 Styles

Le *style* `article` correspond à un ensemble de définitions de variables et de commandes adaptées à la rédaction d'un article. Il existe d'autres styles prédéfinis comme `book`, `report`. On peut aussi *écrire de nouveaux styles* pour des besoins spécifiques.

1.2 Polices de caractères

Par défaut, la taille de la police est de 10 points. Le *point* est l'unité de base pour l'impression. On peut indiquer une autre taille avec un *paramètre optionnel* (entre []). Essayez

```
\documentclass[12pt]{article}
```

On peut localement à l'intérieur d'un document changer la taille de la police au moyen des commandes `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`. Ces commandes peuvent être appliquée sur une partie du texte : `{\Large Titre}` ou porter sur la suite du texte : `\footnotesize` On peut alors revenir à la taille par défaut avec `\normalsize`.

Par défaut, la police de caractère utilisée est roman. Les commandes pour changer de police ont des noms de la forme `\textfn`, $fn \in \{rm, sf, tt, md, bf, up, it, sl, sc\}$, et prennent comme unique argument le texte sur lequel appliquer la police. Exemple : `\textit{Ce texte sera en italique}`. La commande `\emph` permet de mettre en valeur du texte (en italique si le contexte ne l'est pas, pas en italique si le contexte l'est).

Il est possible de combiner certaines de ces commandes (essayez par exemple : `\textbf{\emph{texte}}`, `\textsf{\emph{texte}}`, `\textsf{\textbf{texte}}`). Expérimentez différentes tailles et polices.

1.3 Environnements

Un environnement est une portion de texte positionnée entre deux commandes de la forme :

```
\begin{env}  
...  
\end{env}
```

où `env` est le nom de l'environnement. Essayez l'environnement `verbatim` qui permet taper du texte brut lequel ne sera pas interprété par \LaTeX . Dans la section ?? nous verrons comment définir un nouvel environnement.

1.4 \LaTeX et Emacs

Remarquez qu'Emacs a un mode adapté à \LaTeX . Recherchez une commande et la clé associée qui permet de clore automatiquement par `\end{env}` un environnement ouvert par `\begin{env}`. Explorez les possibilités offertes par le mode `LaTeX`.

2 Réalisation d'un document standard

Téléchargez le fichier `exemple.tex`. Compilez-le.

2.1 Découpage du texte

Remarquez le découpage du texte en *sections*, *sous-sections*, *paragraphes* et la *numérotation automatique*. Entraînez-vous en rajoutant des sections de chaque catégorie.

2.2 Listes

Remarquez l'environnement `itemize`. Testez les environnements similaires `enumerate`, `description`. Pour ce dernier chaque item doit commenter par une description :

```
\item[description de l'item] texte de l'item
```

3 Références croisées

La plupart des entités de votre document (section, figure, page, ...) peuvent être référencés au moyen de *références* croisées. Pour pouvoir référencer une entité il faut lui donner un nom grâce à la commande `\label` dont l'argument correspond au nom qu'on donne à l'entité.

```
\section{Résultats}  
\label{resultats}
```

On peut alors récupérer le numéro de la section au moyen de `\ref{resultats}` comme dans la phrase :

Les résultats sont présentés dans la section~\ref{resultats}.

Le caractère `~` permet d'insérer un petit espace tout en évitant que \LaTeX ne coupe entre le mot « section » et le numéro de la section. On peut également utiliser `\pageref{resultat}` pour obtenir le numéro de la page où se trouve la définition de l'étiquette `resultat`.

Essayez ce mécanisme. Notez que \LaTeX doit être exécuté deux fois pour que les références soient correctes ; la première fois, il sauvegarde les références dans le fichier `exemple.aux`, la deuxième il utilise ces informations pour remplacer les `\ref{...}` par la valeur correcte.

3.1 Table des matières

Rajoutez dans le fichier `exemple.tex` la commande `\tableofcontents`. Celle-ci produit automatiquement une table des matières à l'endroit où la commande est positionnée. Deux passes de \LaTeX sont nécessaires pour obtenir une table correcte. Lors de la première passe \LaTeX crée un fichier `exemple.toc` ; lors de la deuxième passe il utilise l'information de ce fichier pour produire la table.

Testez ce mécanisme.¹

3.2 Bibliographie

Téléchargez le fichier `biblio.bbl`. Inspectez son contenu.

3.2.1 Références et citations

Insérez le contenu du fichier `biblio.bbl` à la fin de votre fichier `exemple.tex` (avant le `\end{document}`). Rajoutez des commandes `\cite{...}` dans votre texte et recompilez une fois votre document.

Remarquez que le résultat contient maintenant la bibliographie mais que les citations contiennent un '?'. Recompilez une deuxième fois et constatez que les références sont maintenant correctes. Pourquoi faut-il compiler 2 fois pour que les citations soient correctes ?

La solution ci-dessus consistant à écrire les références « à la main » n'est pas bonne ; en particulier pour chaque style de bibliographie (souhaitée ou imposée par un éditeur) il faudra réécrire les `bibitem` au format demandé. La bonne solution consiste à utiliser le programme `bibtex` pour produire les `bibitem` dans le style souhaité à partir d'une base de données bibliographique initiale.

3.2.2 Bibtex

Téléchargez le fichier `biblio.bib`. Inspectez son contenu. Dans le fichier `exemple.tex` supprimez la partie concernant la bibliographie insérée précédemment et remplacez-la par les lignes suivantes :

```
\bibliographystyle{plain}
\bibliography{biblio}
```

Recompilez votre fichier `exemple.tex` puis effectuez la commande `bibtex exemple`. Cette dernière a pour effet de créer le fichier `exemple.bbl` à partir du fichier `biblio.bib` et des citations effectuées dans `exemple.tex`. Examinez le contenu du fichier `exemple.bbl`. Recompilez `exemple.tex` et regardez le résultat obtenu.

Remplacez la ligne `\bibliographystyle{plain}` par la ligne `\bibliographystyle{alpha}` pour obtenir un autre style de bibliographie. Refaites `bibtex`, examinez le nouveau fichier `exemple.bbl` obtenu. Puis recompilez `exemple.tex` et regardez le résultat final.²

4 Programmation de \LaTeX

4.1 Définition de commandes

```
\newcommand{\paire}[2]{(#1,#2)}
...
\paire{1}{2}, \paire{2}{3}
```

Modifiez cette définition (par exemple en remplaçant les parenthèses par un un caractère ou en modifiant la police) et testez la modification. En vous inspirant du modèle ci-dessus, définissez une nouvelle commande et testez-la.

Si la commande existe déjà, `\newcommand` produit une erreur. Pour redéfinir une commande existant il faut utiliser `\renewcommand`. Essayez par exemple :

```
{\renewcommand{\labelitemi}{$\heartsuit$}
\begin{itemize}
\item joli item1
\item joli item2
\end{itemize}}
```

¹On peut similairement obtenir automatiquement une liste des figures, une liste des tables, un index.

²Par défaut n'apparaissent dans la bibliographie que les références qui on fait l'objet d'un `\cite` dans l'article. Si on veut qu'une référence apparaisse même si elle n'est pas citée, il faut utiliser `\nocite{...}`.

4.2 Définition de compteurs

```
\newcounter{compteur}
\setcounter{compteur}{0}
première valeur: \thecompteur\
\addtocounter{compteur}{1}
deuxième valeur: \thecompteur\
```

4.3 Définition d'environnement

```
\newenvironment{question}{\noindent{\large \textbf{Question --}}}{Fin\[\[.2cm]}
```

```
\begin{question}
première question
\end{question}
```

```
\begin{question}
deuxième question
\end{question}
```

Améliorez l'environnement `question` précédent en lui associant un compteur qui permettra de numérotter les questions et obtenir le résultat suivant :

Question 1 – première question Fin

Question 2 – première question Fin

Remarque. Pour redéfinir un environnement déjà défini on utilise la commande `\renewenvironment` au lieu de la commande `\newenvironment`.

5 Texte mathématique

L^AT_EX est particulièrement adapté à la rédaction de textes mathématiques. Une expression mathématique insérée dans du texte est entourée de `$` (ou placée dans un environnement `math`). Par exemple :

L'équation de la droite est $3x - 2y + 4 = 0$.

On obtient x^i et x_i respectivement par `x^i` et `x_i`. Pour obtenir une formule centrée sur une ligne, on la place entre `\[` et `\]`.

Au moyen des commandes `\sqrt{x}` (\sqrt{x}), `\frac{a}{b}` ($\frac{a}{b}$), `\sum_{a}^b` (\sum_a^b), `\int_{a}^b` (\int_a^b), `\overline{x}` (\bar{x}) et `\lambda` (λ), écrivez les formules suivantes :

$$z_1 = \sqrt{x_1^2 + y_1^2}$$

$$T(I) = \sum_{i=0}^I \frac{h(i)}{M}$$

$$X = \frac{\int_{\lambda_{min}}^{\lambda_{max}} E(\lambda)S(\lambda)\bar{x}(\lambda)d\lambda}{\int_{\lambda_{min}}^{\lambda_{max}} S(\lambda)\bar{y}(\lambda)d\lambda}$$

Essayez de placer les formules entre `$`, puis entre `\[` et `\]` et observez la différence.